Amendments to the Specification

Please replace the paragraph that begins on Page 20, line 1 and carries over to Page 21, line 5, with the following marked-up replacement paragraph:

-- Fig. 4B illustrates how the present invention supports server-initiated TCP request messages. Suppose Telnet server 381 wishes to send a TCP request to Telnet client 301. Because HTTP is a client-oriented protocol, all messages must originate from the client. The present invention solves this problem by sending an HTTP GET request message (depicted in Fig. 4B as HTTP GET request message 460) from the Web redirector 320 on the receive channel 330, where this HTTP GET request serves to open the communications channel through the redirector servlet 360 to the target server 381. As with the other HTTP messages used in the preferred embodiment, the MIME type of HTTP GET request message 460 is preferably set to "binary/tcp". Web Redirector 320 is always listening on this receive channel 330 for an incoming message from redirector servlet 360. After redirector servlet 360 receives incoming HTTP GET request 460, it performs a TCP read request on the TCP socket for connection 370. This call blocks until the server 381 puts its response onto the TCP socket queue, as in the prior art. When target server 381 has a message to send to client 301, it generates a TCP request 470 and sends that request on TCP connection 370 to redirector servlet 360. Upon receiving TCP request message 470, the redirector servlet's TCP read request on the socket for connection 370 unblocks. Redirector servlet 360 ~~extracts the server's TCP request 470 from the body of the HTTP GET response 480;~~ inserts the TCP request 470 into the body of an HTTP GET response message ~~480;~~ 480 and forwards that HTTP GET response 480 to the Web redirector 320 on receive channel 330. (The MIME type of this HTTP GET response message 480 is also

preferably set to "binary/tcp".) Upon receiving this response 480, thereby completing the round-trip of HTTP GET request 460, the Web redirector 320 (1) sends another HTTP GET request message to redirector servlet 360 (in order to accept the next incoming message generated by target server 381) and (2) extracts the server's TCP request 470 from the body of the HTTP GET response 480. This extracted TCP request is then forwarded to the client 301 on the local TCP connection 310, as shown at 490. This process of listening for incoming messages on the receive channel, extracting the content, and forwarding it to the client then repeats until the connection between client 301 and server 381 is closed.

A'
end

Serial No. 09/619,178   -3-   Docket RSW9-2000-0054-US1